

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
КАФЕДРА ЕКОНОМІЧНОЇ КІБЕРНЕТИКИ

О.М. Густера

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ
до виконання лабораторних робіт з дисципліни
«Інтернет речей»

Галузь знань: 05 «Соціальні та поведінкові науки»
Спеціальність 051 «Економіка»
Спеціалізація «Цифрова економіка»

УДК 519.25:004.67(076.5)
ББК У.с51р
І619

Рецензенти:

Рекомендовано Вченою радою Навчально-наукового інституту Економіки та менеджменту Національного авіаційного університету як методичні рекомендації до виконання практичних завдань з дисципліни «Інтернет речей» для студентів спеціалізації «Цифрова економіка» (протокол № _ від _____ року).

Густера О.М. Інтернет речей [Текст] : методичні рекомендації до виконання лабораторних робіт / О.М. Густера. – К. :НАУ, 2019. – 41 с.

У методичних рекомендаціях представлені практичні завдання в ході виконання яких студенти працюють з основними платформами для побудування та роботи Інтернету речей. Видання призначене для студентів вищих навчальних закладів, докторантів, викладачів та слухачів курсів післядипломної освіти.

СХВАЛЕНО
на засіданні кафедри економічної
кібернетики НАУ і рекомендовано до
друку (протокол № від _____ року)

© О.М. Густера, 2019

© НАУ, 2019

ЛАБОРАТОРНА РОБОТА № 1 ВІРТУАЛЬНЕ ПРОЕКТУВАННЯ ЕЛЕКТРИЧНИХ ПРИЛАДІВ. ЗНАЙОМСТВО З PROTEUS VSM.

Мета роботи: ознайомитись з середовищем Proteus VSM. Оволодіти основними принципами роботи світлодіодів та підключення світлодіодів до живлення.

1. Теоретичні відомості

1.1 Світлодіод та його використання

Світлодіоди, або світловипромінюючі діоди (СВД, в англійському варіанті LED — light emitting diode)— напівпровідниковий прилад, що випромінює некогерентне світло при пропусканні через нього електричного струму. Робота заснована на фізичному явищі виникнення світлового випромінювання при проходженні електричного струму через р-п-перехід. Колір світіння (довжина хвилі максимуму спектра випромінювання) визначається типом використовуваних напівпровідникових матеріалів, що утворюють р-п-перехід.



Переваги:

1. Світлодіоди не мають ніяких скляних колб і ниток розжарювання, що забезпечує високу механічну міцність і надійність (ударна і вібраційна стійкість);
2. Відсутність розігріву і високих напруг гарантує високий рівень електро- і пожежобезпеки;
3. Безінерційність робить світлодіоди незамінними, коли потрібна висока швидкодія;
4. Мініатюрність;
5. Довгий термін служби (довговічність);
6. Високий ККД;

7. Відносно низькі напруги живлення та споживані струми, низьке енергоспоживання;

8. Велика кількість різних кольорів світіння, спрямованість випромінювання;

9. Регульована інтенсивність.

Недоліки:

1. Відносно висока вартість. Відношення ціна/люмен для звичайної лампи накалювання порівняно зі світлодіодами становить приблизно 100 раз;

2. Малий світловий потік від одного елементу;

3. Деградація параметрів світлодіодів з часом;

4. Підвищені вимоги до живлячого джерела.

Зовнішній вигляд та основні параметри:

У світлодіодів є декілька основних параметрів:

1. Тип корпусу

2. Типовий (робочий) струм

3. Зменшення (робоче) напруги

4. Колір свічення (довжина хвилі, нм)

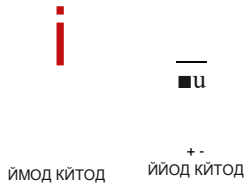
5. Кут розсіювання

Загалом, під типом корпусу розуміють діаметр та колір колби (лінзи). Як відомо, світлодіод - напівпровідниковий прилад, який необхідно живити струмом. Таким чином струм, яким слід живити той чи інший світлодіод називається типовим. При цьому на світлодіоді падає певна напруга. Колір випромінювання визначається як використовуваними напівпровідниковими матеріалами, так і легуючими домішками. Найважливішими елементами, використовуваними в світлодіодах, є: Алюміній (Al), Галій (Ga), Індій (In), Фосфор (P), що викликають свічення в діапазоні від червоного до жовтого кольору. Індій (In), Галій (Ga), Азот (N) використовують для здобуття блакитного і зеленого свічень. Крім того, якщо до кристала, що викликає блакитне (синє) свічення, додати люмінофор, то отримаємо білий колір світлодіода. Кут випромінювання також визначається виробничими характеристиками матеріалів, а також колбою (лінзою) світлодіода.

В даний час світлодіоди знайшли вживання в самих різних областях: світлодіодні ліхтарі, автомобільна світлотехніка, рекламні вивіски, світлодіодні панелі і індикатори, світлофори і т.п.

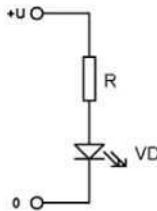
Схема підключення та розрахунок необхідних параметрів:

Оскільки світлодіод є напівпровідниковим приладом, то при включенні в електричне коло необхідно дотримувати полярність. Світлодіод має два виводи, один з яких катод ("мінус"), а інший - анод ("плюс").



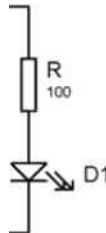
1. Маємо один світлодіод, як його підключити правильно у самому простішому випадку?

Щоб правильно підключити світлодіод у самому простішому випадку, необхідно підключити його через струмообмежувальний резистор.



Задача 1

Є світлодіод з робочою напругою 3 вольти і робочим струмом 20 мА. Необхідно підключити його до джерела з напругою 5 вольт.



Розрахунок опору струмообмежувального резистора виконується за формулою:

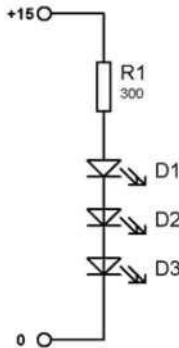
^к "гасяча "світлодіода
"гасяча "живлення "світлодіода

2. Як підключити декілька світлодіодів?

Декілька світлодіодів підключаємо послідовно чи паралельно, розраховуючи необхідні опори.

Задача 2.

Є світлодіоди з робочою напругою 3 вольти і робочим струмом 20 мА. Треба підключити 3 світлодіоди до джерела 15 вольт.

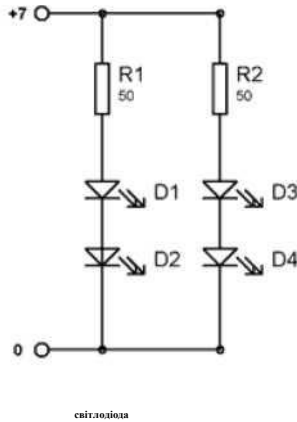


Розрахунок аналогічний попередній задачі:

"гасяча "живлення "світлодіода
"гасяча "живлення "світлодіода
"світлодіода

Задача 3.

Нехай є світлодіоди з робочою напругою 3 вольти і робочим струмом 20 мА. Треба підключити 4 світлодіоди до джерела 7 вольт.



$$I = I - N * U \cdot$$

гасача живлення

світлодіода

1.2 Середовище віртуального проектування електронних схем Proteus VSM

PROTEUS VSM - середовище для проектування та симуляції роботи електронних схем. Відмінністю пакета Proteus VSM є можливість моделювання роботи програмованих пристроїв: мікроконтролерів (МК), мікропроцесорів, DSP та ін..

PROTEUS VSM дозволяє дуже достовірно моделювати і налагоджувати досить складні пристрої, в яких може міститися кілька МК одночасно і навіть різних родин в одному пристрої.

PROTEUS складається з двох основних модулів:

- ISIS - графічний редактор принципів схем служить для введення розроблених проектів з подальшою імітацією і передачею для розробки друкованих плат в ARES. До того ж після налагодження пристрою можна відразу розвести друковану плату в ARES яка підтримує авто розміщення і трасування по вже існуючій схемі.

- ARES - графічний редактор друкованих плат з вбудованим менеджером бібліотек і автотрасувальник ELECTRA, автоматичної розстановкою компонентів на друкованій платі.

На рисунку 1.1 зображене головне вікно програми. Увесь робочий простір програми розподілено на декілька основних частин:

- вікно редактору схем (виконується синтез з окремих компонентів);

- вікно вибору об'єктів (доступні різні елементи в залежності від

вибраного режиму);

- панель керування симуляцією (знаходиться у лівому нижньому куті, включає в себе такі команди: пуск; виконання одного такту, яке включає симуляцію на час Single Step Time, який задається у розділі головного меню System>Set Animation Options; пауза; стоп).

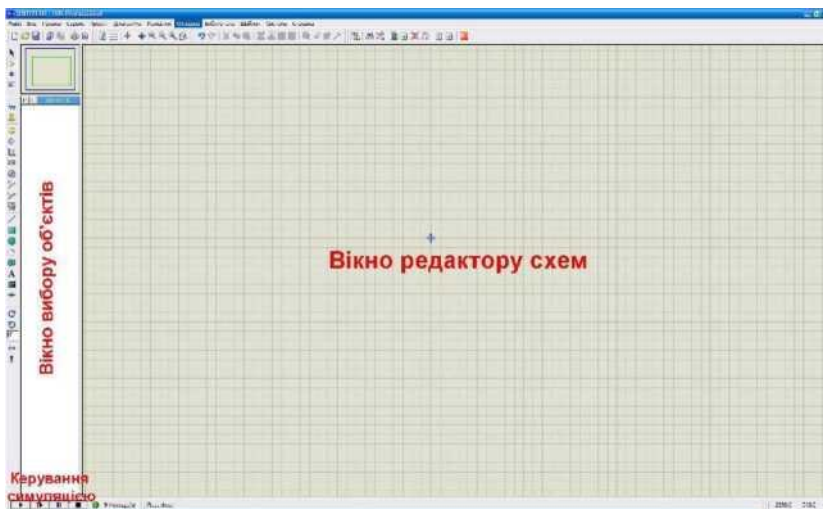


Рисунок 1.1 - Головне вікно Proteus VSM (ISIS)

1.3 Проектування віртуальної моделі електронної схеми та симуляція її роботи в Proteus VSM

Для того щоб зібрати схему будь-якого пристрою, необхідно підготувати набір елементів, з яких буде складатися ця схема. Для цього переходимо в режим компонентів (рис. 1.2а) і клікаєм клавішу P, яка знаходиться під вікном перегляду поряд з клавішею L (рис 1.2б).



Рисунок 1.2 - Панель інструментів

Перед нами з'являється менеджер компонентів, який надає на наш вибір всі елементи, які містяться в бібліотеці програми (рис 1.3).

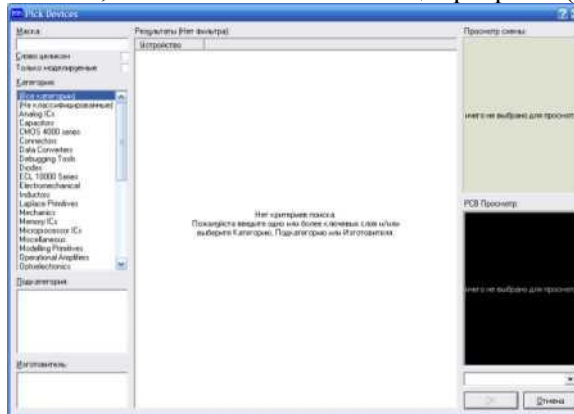


Рисунок 1.3 - Вікно менеджера компонентів

Треба користуватися рядком пошуку, який знаходиться у верхньому лівому куті. Коли потрібний компонент знайдений, двійним кліком лівої кнопки миші по його назві додасть його до переліку використуваних компонентів. Наприклад створимо електрону схему яка буде складатися з: світлодіода, резистора, кнопки і блоку живлення (батареї). У рядку пошуку (маска) вводимо перший елемент який будемо додавати до схеми: led-green. У списку елементів з'являється світлодіод з такою назвою (рис.1.4), клікаємо по ньому двійним кліком лівою кнопкою миші.



Рисунок 1.4 - Пошук компонентів

Далі так само робимо пошук за такими назвами: resistor, push button, battery (бібліотека - Active), - і по черзі двійним кліком по цих елементах у списку додаємо їх до нашого проекту. Після того як ми вибрали всі компоненти натискаємо на кнопку ОК. Тепер наше вікно вибору об'єктів буде містити всі чотири компоненти що ми вибрали і виглядати як зображено на рисунку 1.5.

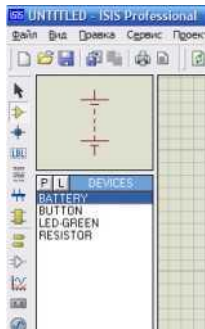
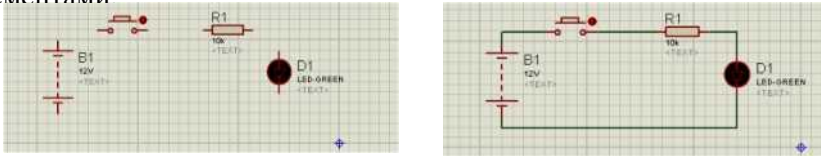


Рисунок 1.5 - Вікно вибору об'єктів

Для встановлення компоненту у вікні редактора схеми його необхідно вибрати зі списку і двійним кліком лівої кнопки миші встановити у потрібному місці. До того як встановити компонент на схемі його можна попередньо розвернути у потрібне положення, яку можна контролювати у вікні перегляду.

Встановіть всі компоненти як зображено на рисунку 1.6 (а), та з'єднайте їх (рис 1.6 б). Для того щоб з'єднати два компоненти між собою для початку виберіть інструмент Стрілка, потім на самій схемі

наведіть курсор миші на кінець елемента, повинен з'явитися квадратний червоний контур, натискаємо і відпускаємо ЛКМ, і ведемо курсор до контакту іншого елемента, і на кінці іншого елемента також натискаємо ЛКМ, щоб завершити побудову з'єднання між двома елементами



а) б)

Рисунок 1.6 - Побудова електричної схеми

Після того як електричне коло створено, міняємо характеристики наших компонентів. Зараз батарея живить схему 12 В, а нам потрібно змінити живлення схеми на 5 В. Для цього клікаємо двійним кліком ЛКМ на батареї і у полі Voltage встановлюємо 5V.

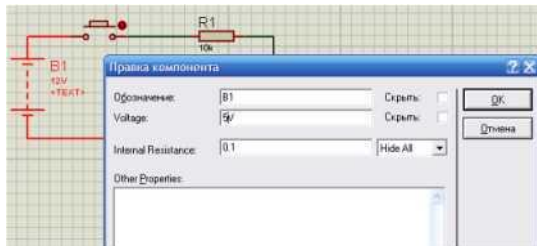


Рисунок 1.7 - Змінення параметрів живлення

Так само змінюємо опір резистора на 680 Ом.

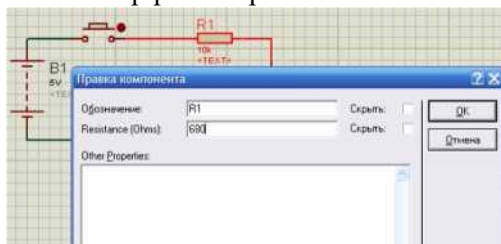


Рисунок 1.8 - Змінення параметрів опору

Після натискаємо в панелі керування симуляцією кнопку «Воспроизвести».

Воспроизвести |----- 1 |
 || к П |» і 1-й П |

Рисунок 1.9 - Панель керування симуляцією

Тепер наша схема може віртуально відтворити включення та виключення світлодіода при натисканні на кнопку (рис 1.10). Для цього просто наведіть курсор миші на елемент кнопка у схемі і натисніть ЛКМ.

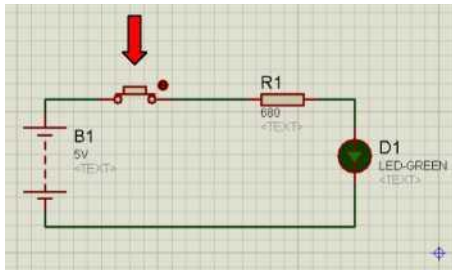


Рисунок 1.10 - Симуляція роботи світлодіода

2. Завдання до лабораторної роботи

- 2.1 Спроекувати приклад приведений у розділі 1.3.
- 2.2 Вирішити 3 задачі які наводяться у теоретичній частині цієї роботи.
- 2.2 Спроекувати віртуальні моделі електронних схем та симуляція їх роботи в Proteus VSM.
- 2.3 Оформити звіт

3. Зміст звіту

- 3.1 Тема та мета роботи
- 3.2 Рішення 3 задач
- 3.3 Схема електрична-принципова підключення світлодіода у середовищі Proteus VSM
- 3.4 Висновки з роботи

ЛАБОРАТОРНА РОБОТА № 2

СТВОРЕННЯ ПРОЕКТУ РОБОТИ ЗІ СВІТЛОДІОДАМИ З ВИКОРИСТАННЯМ СЕРЕДОВИЩ PROTEUS VSM ТА ATMEL STUDIO

Мета роботи: ознайомитись з середовищами Proteus VSM та Atmel Studio 6.2. Навчитися створювати проект віртуальної симуляції роботи світлодіода. Ознайомитись з різновидами функцій та команд для створення прототипу світлофора.

1. Теоретичні відомості

1.1 Створення проекту програми для мікроконтролера у Atmel Studio 6.2

Atmel Studio - інтегроване середовище розробки (IDE) для розробки 8-ми і 32-х бітних додатків від компанії Atmel, що працює в операційних системах Windows NT/2000/XP/Vista/7. Atmel Studio містить асемблер і симулятор, що дозволяє відстежити виконання програми. Atmel Studio містить в собі менеджер проектів, редактор вихідного коду, інструменти віртуальної симуляції та внутрішньосхемного налагодження, дозволяє писати програми на асемблері або на C / C ++.

Після того як ви встановите *Atmel Studio 6.2* на свій комп'ютер, створіть новий проект:

1. Відкрийте Atmel Studio 6.2 та виберіть **New Project**. У діалоговому вікні New Project, виберіть **GCC C Executable Project** як шаблон для створення C коду (рис.2.1). Введіть ім'я проекту і вкажіть місце де він буде зберігатися. Назвемо наш проект "BlinkLED". Зніміть прапорець **Create directory for solution** для спрощення структури каталогів проекту. Клікніть **ОК**.

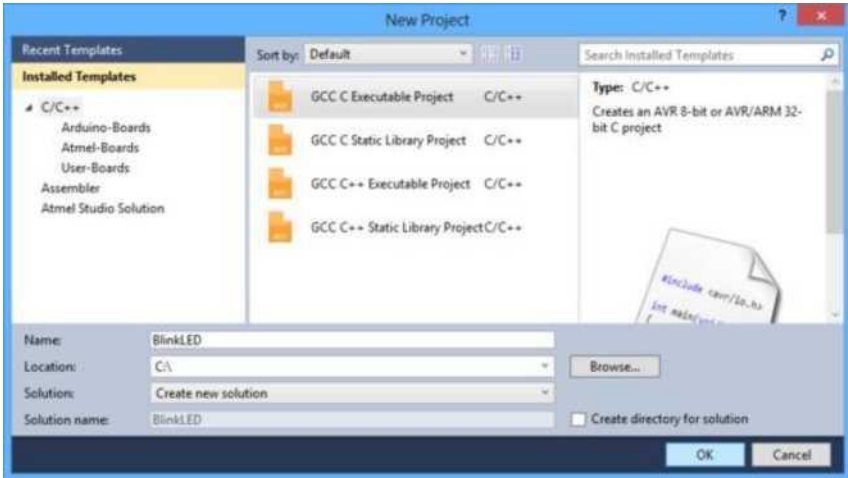


Рисунок 2.1 - Діалогове вікно New Project в Atmel Studio 6

2. У вікні Device Selection (рис. 2.2), оберіть імя пристрою AVR яке необхідно. Клікніть **ОК** для створення нового проекту.

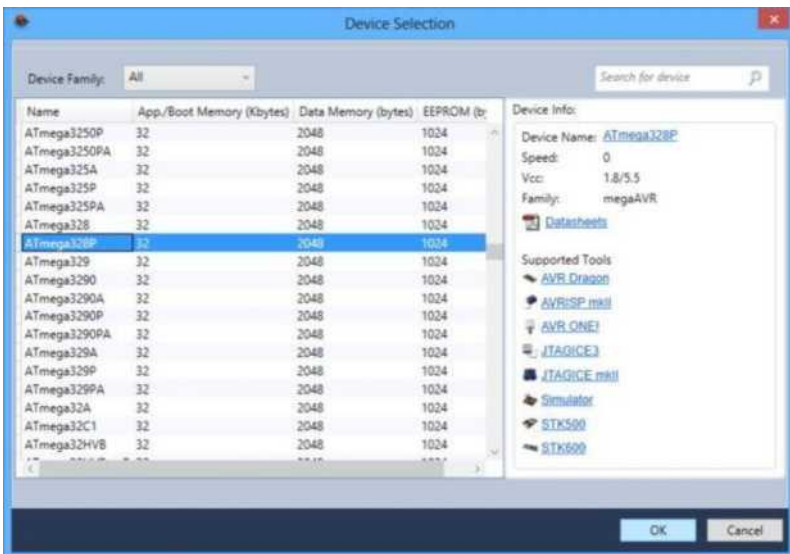


Рисунок 2.2 - Діалогове вікно Device Selection в Atmel Studio 6

3. Напишіть код у BlinkLED.cpp приведений нижче:

```

#define F_CPU 8000000 // частота AVR в Гц,
використовується для util/delay.h
#include <avr/io.h>
#include <util/delay.h> int main() {
    DDRD |= (1<<DDD1); // установка LED виходу PD1 на
вихід
    while (1) {
        PORTD |= (1<<PORTD1); // PD1 включення
        _delay_ms(50); // затримка на 50 мс
        PORTD &= ~(1<<PORTD1); // PD1 виключення
        _delay_ms(100); // затримка 100 мс
    }
}

```

4. На панелі інструментів клікніть кнопку **Build Solution** (або натисніть **F7**), щоб скомпілювати код (рис.2.3).

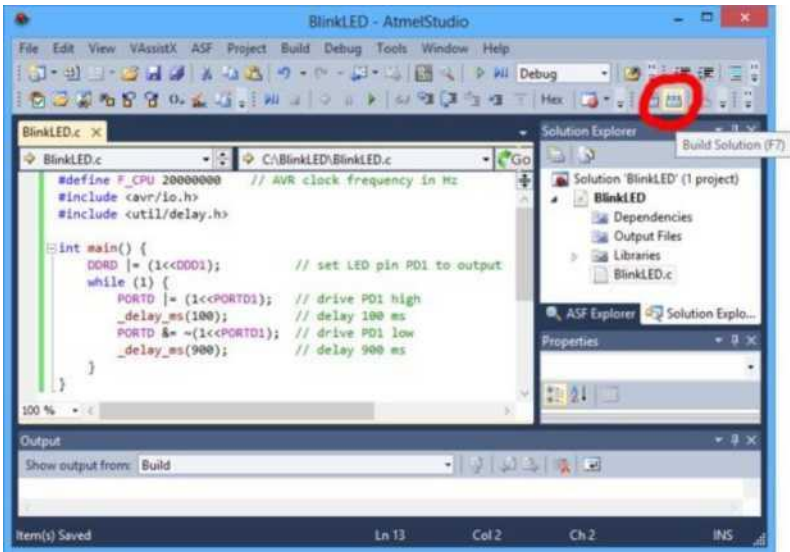


Рисунок 2.3 - Побудова проекту

Після компіляції перейдіть в директорію проекту і знайдіть файл BlinkLED.hex - це файл, який представляє собою довільні двійкові данні у текстовому вигляді. По історичним причинам є стандартом де-факто при прошивці різноманітних мікросхем. Саме його необхідно буде завантажити у мікроконтролер схеми у Proteus VSM, яка буде побудована далі.

1.2 Симуляція роботи мікроконтролера за допомогою Proteus VSM


На рисунку 2.4 показане розведення виводів мікроконтролера ATmega328P. Світлодіод ми будемо підключати до третього виводу мікроконтролера (PD1).

(RESET) POBE 1	as <input type="checkbox"/> PCS (ADC5/SCL)
(RXD)PDOE 2	27 <input type="checkbox"/> PC4 (ADC4/SDAJ
(TXD) PD1 E 3	as <input type="checkbox"/> PCS (ADC3)
(INT0)PD2E 4	as <input type="checkbox"/> PC2 (ADCa)
(INT1)PD3E 5	24 2 PC 1 (ADC1)
(XCKT0)PD4E 6	33 <input type="checkbox"/> PC0 (ADCO)
VCCE 7	22 JAGND
GND E 8	21 TAREF
(XTAL1/TOSC1) PBBE 9	20 JAVCC
(XTAL2/TOSC2) PB7 E 10	12 JPB5(SCK) ia 1 PBd
(T1)PD5E 11	(Miso:i
(AIN0)PDBE 12	17 J PBS (MDSI/OC2)
(AIN1)PD7E 13	16 <input type="checkbox"/> PB2(SSOC1B)
(ICPIPBOE 14	15 1PB1 (OC1A)
PDIP	

Рисунок 2.4 - Розташування виводів мікроконтролера ATmega328P

Створіть новий проект у Proteus VSM. І розкресліть схему, як зображено на рисунку 2.5. У менеджері компонентів знайдіть нижче приведені елементи:

1. світлодіод
2. мікроконтролер ATmega328P
3. резистор
4. заземлення

Для того щоб створити елемент «заземлення» (GROUND) перейдіть у панелі інструментів (знаходиться вертикально зліва) на вкладку Terminals Mode ().

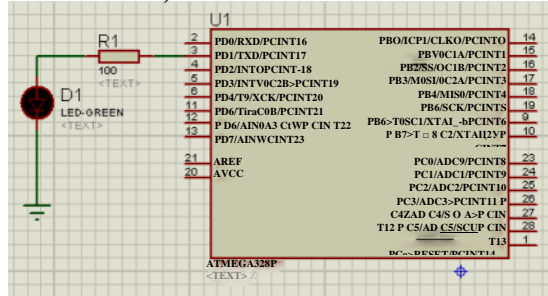


Рисунок 2.5 - Схема підключення

Після з'єднання елементів у електричне коло, двійним кліком клікніть на зображенні мікроконтролера. Відкриється вікно Edit Component (рис. 2.6), в якому у полі Program File вказуємо шлях до файлу BlinkLED.hex. також перевірте щоб була встановлена частота мікроконтролера така сама як і у програмному кодї, тобто 8МГц.

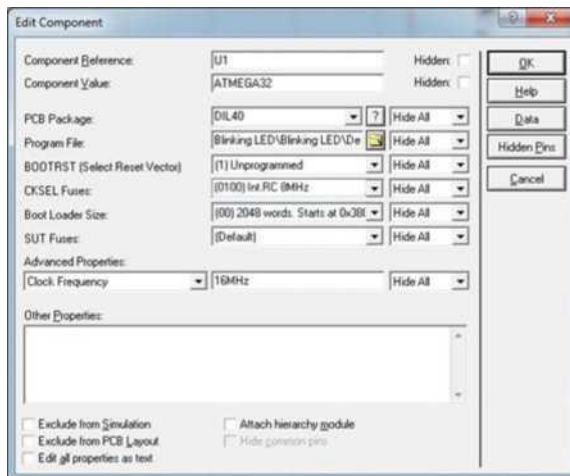
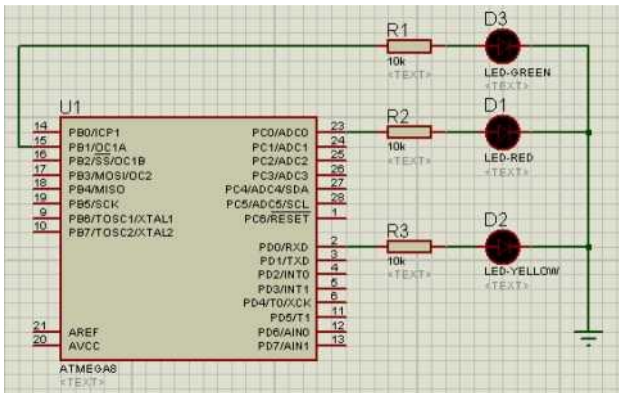


Рисунок 2.6 - Вікно Edit Component

Після на панелі керування симуляцією клікаємо на Play, і якщо все правильно було зроблено, споглядаємо роботу схеми.

2. Завдання до лабораторної роботи

2.1 Розробіть програму і виконати симуляцію мерехтіння трьох різних світлодіодів підключених до мікроконтролера ATmega8. Схема підключення у середовищі Proteus VSM приведена нижче:



3. Зміст звіту

- 3.1. Тема та мета роботи
- 3.2. Схема електрична-принципова
- 3.3. Код програми
- 3.4. Висновки з роботи

ЛАБОРАТОРНА РОБОТА № 3 ЗНАЙОМСТВО З ARDUINO. ПРИНЦИПИ РОБОТИ ЗІ СВІТЛОДІОДАМИ

Мета роботи: ознайомитись з платою Arduino. Оволодіти основними принципами роботи та підключення світлодіодів до плати, ознайомитись з різновидами функцій та команд для створення прототипу світлофора.

1. Теоретичні відомості

1.1 Arduino

Arduino - це електронний конструктор і зручна платформа швидкої розробки електронних пристроїв для новачків і професіоналів. Переваги платформи: зручна у використанні, простота мови програмування, а також відкрита архітектура і програмний код. Пристрій програмується через USB без використання програматорів.

Пристрої на базі Arduino можуть отримувати інформацію про навколишнє середовище за допомогою різних датчиків, а також можуть управляти різними виконавчими пристроями.

Існує декілька версій платформ Arduino. В цьому курсі буде розглядатися платформа Arduino Uno (рис. 3.1).



Рисунок 3.1 - Зовнішній вигляд Arduino Uno

Arduino Uno базується на мікроконтролері Atmel ATmega328. Мікроконтролер на платі програмується за допомогою мови Arduino (заснований на мові Wiring) і середовища розробки Arduino (заснована на середовищі Processing). Проекти пристроїв, засновані на Arduino, можуть працювати самостійно, або ж взаємодіяти з програмним забезпеченням на комп'ютері (напр.: Flash, Processing, MaxMSP). Плати можуть бути зібрані користувачем самостійно або куплені в зборі. Програмне забезпечення доступне для безкоштовного скачування. Вихідні креслення схем (файли CAD) є загальнодоступними, користувачі можуть застосовувати їх на свій розсуд.

Платформа має 14 цифрових вхід/виходів (6 з яких можуть використовуватися як виходи ШІМ), 6 аналогових входів, кварцовий

генератор 16 МГц, роз'єм USB, силовий роз'єм, роз'єм ICSP і кнопку перезавантаження. Кожен з 14 цифрових виводів Uno може налаштований як вхід або вихід, використовуючи функції `pinMode ()`, `digitalWrite ()`, і `digitalRead ()`. Кожен вивід має навантажувальний резистор (за замовчуванням відключений) 20-50 кОм і може пропускати до 40 мА. Для роботи необхідно підключити платформу до комп'ютера за допомогою кабелю USB, або подати живлення за допомогою адаптера AC/DC або батареї.

Платформа може працювати при зовнішньому живленні від 6 В до 20 В. При напрузі живлення нижче 7 В, вивід 5V може видавати менше 5 В, при цьому платформа може працювати нестабільно. При використанні напруги вище 12 В регулятор напруги може перегрітися і пошкодити плату. Рекомендований діапазон від 7 В до 12 В.

Деякі виводи мають особливі функції:

- VIN. Вхід використовується для подачі живлення від зовнішнього джерела (в відсутність 5 В від роз'єму USB або іншого регульованого джерела живлення). Подача напруги живлення відбувається через даний вивід.

- 5V. Регульоване джерело напруги, що використовується для живлення мікроконтролера і компонентів на платі. Живлення може подаватися від виводу VIN через регулятор напруги, або від роз'єму USB, або іншого регульованого джерела напруги 5 В.

- 3V3. Напруга на виводі 3.3 В генерується вбудованим регулятором на платі. Максимальне споживання струму 50 мА.

- GND. Виводи заземлення.

- Послідовна шина: 0 (RX) і 1 (TX). Виводи використовуються для отримання (RX) і передачі (TX) даних TTL. Дані виводи підключені до відповідних роз'ємів мікросхеми послідовної шини ATmega8U2 USB-to-TTL.

- Зовнішнє переривання: 2 і 3 Дані висновки можуть бути налаштовані на виклик переривання або на молодшому значенні, або на передньому чи задньому фронті, або при зміні значення. Детальна інформація знаходиться в описі функції `attachInterrupt ()`.

- ШІМ: 3, 5, 6, 9, 10, і 11. Будь-який з виводів забезпечує ШІМ з роздільною здатністю 8 біт за допомогою функції `analogWrite ()`.

- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). За допомогою даних виводів здійснюється зв'язок SPI, для чого використовується

бібліотека SPI.

- LED: 13. Вбудований світлодіод, підключений до цифрового виводу 13. Якщо значення на виводі має високий потенціал, то світлодіод горить.

На платформі Uno встановлені 6 аналогових входів (позначених як A0 .. A5), кожен роздільною здатністю 10 біт (тобто може приймати 1024 різних значення). Стандартно виводи мають діапазон вимірювання до 5 В відносно землі, проте є можливість змінити верхню межу за допомогою виводу AREF і функції `analogReference()`. Деякі виводи мають додаткові функції:

- I2C: 4 (SDA) і 5 (SCL). За допомогою виводів здійснюється зв'язок I2C (TWI), для створення якої використовується бібліотека `Wire`.

Додаткова пара виводів платформи:

- AREF. Опорна напруга для аналогових входів. Використовується з функцією `analogReference()`.

- Reset. Низький рівень сигналу на виводі перезавантажує мікроконтролер. Звичайно застосовується для підключення кнопки перезавантаження на платі розширення, що закриває доступ до кнопки на самій платі Arduino.

1.2 Основні функції для роботи зі світлодіодами

Для роботи зі світлодіодом необхідно знати і вміти володіти такими функціями і константами:

- оператор `setup()`;
- оператор `loop()`;
- функція `pinMode()`;
- функція `digitalWrite()`;
- функція `delay()`;
- константи `OUTPUT`, `HIGH`, `LOW`.

Далі наведений код програми найпростішого прикладу мерехтіння вбудованим у плату Arduino світлодіодом, який підключено до 13 виводу:

```
void setup()
{ pinMode(13, OUTPUT);
}
```

Ця функція виконується на початку роботи програми (після запуску мікроконтролера). Тобто послідовно виконується кожна команда, яка знаходиться між фігурними скобками цієї функції. Наприкінці кожної строки необхідно поставити символ закінчення команди “;”. Тут функція `setup` містить одну єдину команду - `pinMode(13, OUTPUT)`. Ця команда налаштовує 13 порт Arduino, як вивід. Порт № 13 знаходиться на верхній колодці портів Arduino.

Після функції `setup` виконується функція `loop`.

```
void loop()
{
digitalWrite(13, HIGH); // вмикаємо світлодіод delay(1000); //
чекаємо секунду
digitalWrite(13, LOW); // вимикаємо світлодіод delay(1000); //
чекаємо секунду
}
```

На відміну від `setup`, функція `loop` постійно повторюється - як тільки послідовно виконані всі команди в скобках, функція запускається знову. Функція `loop` для цього прикладу складається з чотирьох команд:

На порт 13 подається напруга (5 В) - світлодіод вмикається.

Затримка до виконання наступної команди 1000 мілісекунд (одну секунду)

Порт 13 з'єднується з землею - світлодіод вимикається.

Ще одна затримка на 1 секунду.

Після виконання усіх чотирьох команд, знову виконується перша команда (включення світлодіоду) и так продовжується до тих пір, поки Arduino включена або поки не буде натиснута кнопка RESET.

2. Завдання до лабораторної роботи

2.2 Підключення світлодіодів до плати Arduino згідно зі схемою, яка зображена на рисунку 3.2.

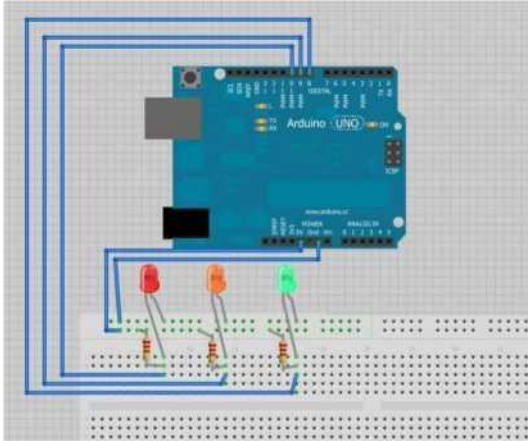


Рисунок 3.2 - Схема підключення світлодіодів

2.3 Запуск на комп'ютері середовища розробки Arduino

2.4 Написання коду згідно з алгоритмом представленим на
рисунок 3.3.

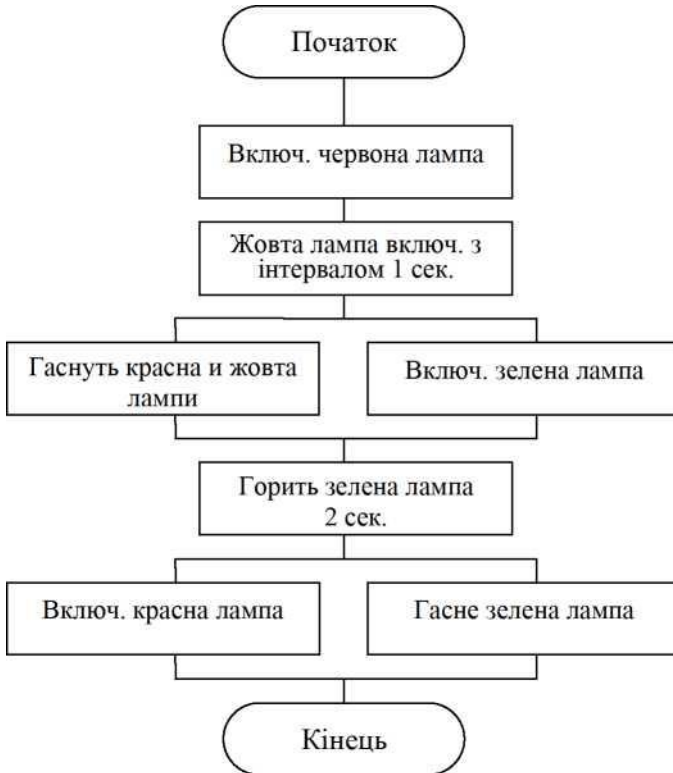


Рисунок 3.3 - Схема алгоритму роботи світлофора

2.5 Підключення плати Arduino через USB до ПК.

2.6 Натисніть кнопку Verify и переконайтесь, що у нижній частині вікна з'явився надпис Done Compiling. Це значить, що у написаній програмі не знайдено помилок.

2.7 Виберіть у Tools->Board ваш тип плати. Перевірте, чи правильно обрано USB-порт в Tools->Serial port. Після натисніть на кнопку Upload.

2.8 Якщо внизу з'явився надпис "Done uploading" - процес запису прошов успішно.

3. Зміст звіту

3.1. Тема та мета роботи

3.2. Схема електрична-принципова світлофору

3.3. Код програми

3.4. Висновки з роботи

ЛАБОРАТОРНА РОБОТА № 4 РОБОТА З КНОПКАМИ

Мета роботи: ознайомитись з основними принципами роботи та підключення кнопки до плати, виконати підключення кнопки до плати Arduino.

1. Теоретичні відомості

Кнопка - найпростіший електромеханічний пристрій, для передачі електричного сигналу різним пристроям шляхом замикання або розмикання двох і більше контактів. В залежності від стану, в якому залишається кнопка після натискування на неї, вона буває двох видів:

- кнопки (перемикачі) з фіксованим положенням;
- тактові кнопки або нефіксовані (які повертаються у вихідне положення).

На рисунку 4.1 зображена схема правильного підключення кнопки до піна плати.

+5V

до порту вводу/виводу плати



Рисунок 4.1 - Схема підключення кнопки

Пін плати на вході повинен мати стан 0 або 1. Коли стан «підвішений», тобто не визначений, на вході будуть збиратися різні зовнішні наводки (статичні, електричні, електромагнітні), що буде призводити до хибного спрацювання кнопки. Щоб підвести пін до 0

або 1 використовують підтягуючі резистори. Вони бувають двох видів, верхньої або нижньої підтяжки. Верхні підключаються до плюса живлення, а нижні до мінуса.

Після підключення схеми переходимо до написання програми. Необхідно пін, до котрого буде підключатися кнопка, ініціалізувати на вхід за допомогою команди:

`pinMode(buttonPin, INPUT);` //де buttonPin - це номер піна до якого підключена кнопка

Також необхідно об'явити змінну в яку буде записуватися статус кнопки.

Розглянемо простий приклад вмикання та вимикання світлодіода за допомогою кнопки, підключеної до 7 піна. Схема підключення зображена на рисунку 4.2.

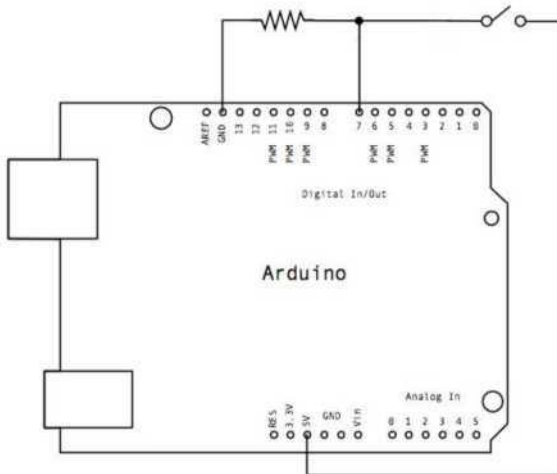


Рисунок 4.2 - Схема підключення кнопки до Arduino Uno

Текст програми, у якій за допомогою натискання кнопки відбувається включення вбудованого світлодіода на виводі 13 виглядає так:

```
const int buttonPin = 7; // номер піна, до якого підключена кнопка
int buttonState = 0; // змінна для запису стану кнопки
void setup ()
pinMode(buttonPin, INPUT); // ініціалізація піна на введення
```

```

void loop ()
{
  buttonState = digitalRead(buttonPin);
  if (buttonState == HIGH) // якщо кнопка натиснута
    digitalWrite(13, HIGH); // вбудований світлодіод вмикається
  }
  else {
    digitalWrite(13, LOW); // вбудований світлодіод вимикається
  }
}

```

3. Завдання до лабораторної роботи

- 2.1 Виконайте підключення до 5 порту світлодіода.
- 2.2 Підключіть до 6 порту кнопку.
- 2.3 Напишіть програму, щоб при натисканні кнопки світло діод, підключений до 5 порту, вмикався, а вбудований світлодіод до 13 порту вимикався.

3. Зміст звіту

- 3.1. Тема та мета роботи
- 3.2. Схема електрична-принципова
- 3.3. Код програми
- 3.4. Висновки з роботи

ЛАБОРАТОРНА РОБОТА № 5 SERVO MOTOR

Мета роботи: ознайомитись з особливістю будови та різновидами серводвигунів, та отримати навички роботи Arduino з серводвигунами.

1. Теоретичні відомості

Сервопривід - привід з управлінням через негативний зворотній зв'язок, що дозволяє точно керувати параметрами руху. Сервопривід з мотором (серводвигун), призначений для приведення в рух пристроїв керування через поворот вихідного валу.

Сімейство серводвигунів різноманітне, його можна розподілити

на декілька груп (рис.5.1). Відміні особливості обумовлені наступними факторами:

- конструкція двигунів (статор, ротор);
- необхідні системи регулювання;
- система зворотного зв'язку (датчики).



Рисунок 5.1 - Огляд серводвигунів

Будова сервоприводу для радіокерованих моделей зображена на рисунку 5.2. Він складається з електродвигуна, укладеного в один корпус з редуктором і керуючою електронікою, яка найчастіше складається з потенціометра зворотного зв'язку та плати управління. Серводвигун використовує три приводи для роботи. Плюсовий привід для живлення, зазвичай 4.8В-6В, мінусовий привід та сигнальний привід. Керуючий сигнал передає інформацію щодо положення вихідного валу. Вал пов'язаний з потенціометром, який визначає його положення. Контролер згідно опору потенціометра і значенням керуючого сигналу визначає, в який бік потрібно обератит мотор, щоб отримати потрібна положення вихідного валу.



Рисунок 5.2 - Конструкція сервоприводу

Для розглядання прикладу керування положенням сервоприводу за допомогою Arduino необхідна плата та сервопривід. Схема підключення зображена на рисунку 5.3. Як вже було сказано вище серводвигун має 3 приводи: живлення, заземлення, сигнальний. Дріт живлення, як правило, червоний, і повинен бути з'єднаний з піном 5V на платі Arduino. Дріт заземлення, як правило, чорного або коричневого кольору і повинен бути підключений до заземлення на платі Arduino. Сигнальний контакт, як правило, жовтого, білого або помаранчевого кольору, і повинен бути підключений до контакту 9 на платі Arduino.

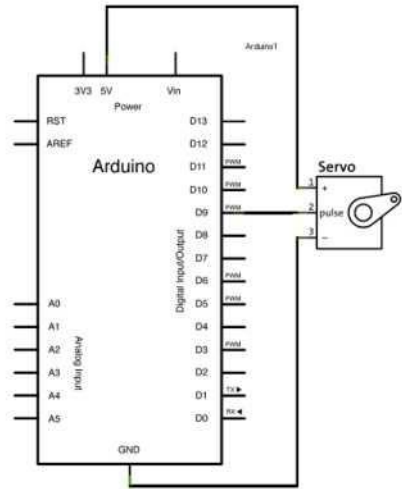


Рисунок 5.3 - Схема підключення сервоприводу до Arduino

Для написання програми у середовищі Arduino є спеціальна бібліотека для роботи з сервоприводом «Servo.h». Далі наведено простий приклад програми роботи з сервоприводом: `#include <Servo.h>`

```
Servo myservo; // створення об'єкта сервоприводу для курування серво;
максимум можна створити 8 об'єктів сервоприводів
```

```
int pos = 0; // змінна для збереження положення сервоприводу
```

```
void setup()
```

```
{
myservo.attach(9); // призначити серво який підключений до 9 піна до
об'єкту сервоприводу
}
```

```
void loop()
```

```
{
for(pos = 0; pos < 180; pos += 1) // рух від 0 до 180 градусів
```

```
{
// с кроком в 1 градус
```

```

myservo.write(pos);           // дає команду сервоприводу перейти на
позицію 'pos'
delay(15);                    // затримка 15мс для того щоб серво досягла
необхідної позиції
}
for(pos = 180; pos>=1; pos-=1) // рух від 180 до 0 градусів
{
  myservo.write(pos);         // дає команду сервоприводу перейти на
позицію 'pos'
  delay(15);                  // затримка 15мс для того щоб серво досягла
необхідної позиції
}

```

2. Завдання до лабораторної роботи

2.1 Підключити кнопку і сервопривід до плати Arduino. Написати програму, щоб при натисканні кнопки сервопривід рухався за годинниковою стрілкою. При відпусканні кнопки сервопривід зупинявся.

2.2 Підключити кнопку і сервопривід до плати Arduino. Написати програму, щоб при натисканні кнопки сервопривід рухався проти годинникової стрілки. При відпусканні кнопки сервопривід зупинявся.

3. Зміст звіту

- 3.1. Тема та мета роботи
- 3.2. Схема електрична-принципова
- 3.3. Код програми
- 3.4. Висновки з роботи

ЛАБОРАТОРНА РОБОТА № 6 ПРОГРАМУВАННЯ ПОТЕНЦІОМЕТРА

Мета роботи: ознайомитися з принципами роботи потенціометра, виконати підключення потенціометра до плати Arduino та розробити програму.

1. Теоретичні відомості

Потенціометр - це змінний резистор з регульованим опором. Потенціометри використовуються в робототехніці як регулятори різних параметрів - гучність звуку, потужності, напруги тощо.

Потенціометр має три контакти (рис. 6.1). Середній контакт йде на аналоговий вихід. Зовнішні контакти йдуть до пiна із живленням і до землі.



Рисунок 6.1 - Зображення потенціометра

Для детального розуміння роботи потенціометра розглянемо приклад його роботи.

На рисунку 6.2 зображена схема підключення потенціометра до плати. У цьому випадку потенціометр підключений між землею і +5 V потенціалами (оскільки плата Arduino живиться +5 V), а движок сполучений з каналом аналого-цифрового перетворювача мікроконтролера. У такому разі можна регулювати вихідну напругу потенціометра в межах від 0 до 5 V.

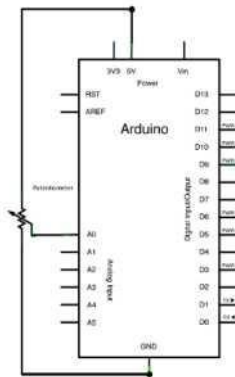


Рисунок 6.2 - Схема підключення потенціометра до плати Arduino

Код програми для роботи потенціометра:

```
void setup() {  
    Serial.begin(9600);  
}  
  
void loop() {  
    int sensorValue = analogRead(A0); // отримуємо поточне  
    значення  
    Serial.println(sensorValue, DEC); // виводимо  
    результат на монітор  
}
```

При ініціалізації встановлюємо потрібну швидкість зв'язку: **Serial.begin(9600)**; Далі в циклі ми постійно зчитуємо дані, що поступають з потенціометра за допомогою методу `analogRead()`. Оскільки значення знаходитимуться в діапазоні від 0 до 1023, можемо використовувати тип `int` для змінної `sensorValue`.

Отриманий результат будемо виводиться у вікно послідовного монітора в десятковому форматі.

2. Завдання до лабораторної роботи

2.1 При повороті ручки потенціометра змінювати яскравість світлодіоду. Схема підключення зображена на рисунку 6.2.

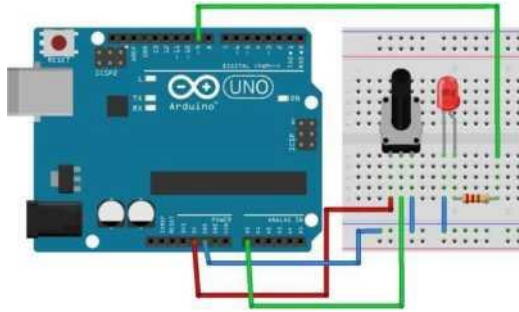


Рисунок 6.2 - Схема підключення

2.2 Використати потенціометр для керування повороту серво.
Схема підключення зображена на рисунку 6.3.

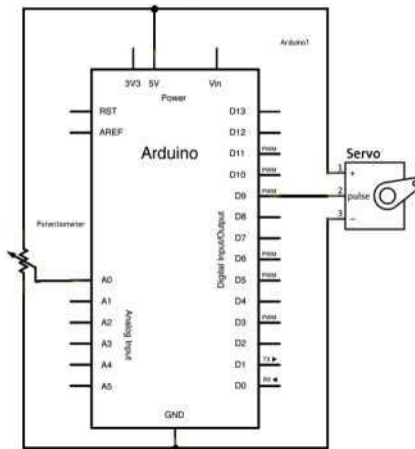


Рисунок 6.3 - Схема підключення

3. Зміст звіту

- 3.1. Тема та мета роботи
- 3.2. Схема електрична-принципова
- 3.3. Код програми
- 3.4. Висновки з роботи

ЛАБОРАТОРНА РОБОТА № 7 СИМУЛЯЦІЯ РОБОТИ ARDUINO ЗА ДОПОМОГОЮ WEB-ДОДАТКУ 123D CIRCUITS

Мета роботи: ознайомитися з web-додатком віртуальної імітації роботи платформи Arduino, розробити свій проект з використанням можливостей web-додатка.

1. Виконання роботи

Сервіс 123D Circuits компанії Autodesk дає можливість створити електронні схеми. Веб-додаток має підтримку платформи Arduino та дозволяє редагувати код, а також дає можливість у візуальному режимі будувати схем і виконувати інтерактивну імітацію їх роботи у реальному часі. Працювати над схемами можна спільно з іншими людьми, використовуючи бібліотеку компонентів.

Для роботи з сервісом 123D Circuits перейдіть за посиланням <http://123d.circuits.io/>.

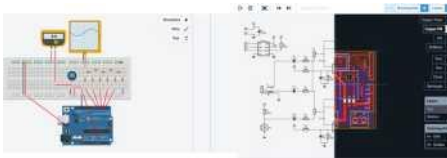
Після проходження реєстрації в 123D Circuits ви опинитесь на головній сторінці (рис. 7.1). Домашня сторінка показує список схем, які ви створили, а також нещодавно

Shop

створені/редаговані.

123DCircuits / Circuits Components Circuit Scribe Blog Help

Your projects are now here



Breadboard Circuits + Simulation

Schematic + PCB

New Breadboard Circuit

New Schematic Circuit

Circuit Scribe Sketch + Simulation

New Circuit Scribe

electrOn

Easily create and simulate
Circuit Scribe sketches online.



Рисунок 7.1 - Головна сторінка 123D Circuits

Circuit Scribe Editor Beta

Щоб створити нову схему перейдіть на вкладку Circuit верхнього меню. Після, натисніть кнопку "Create new circuit" на правій панелі вибору. Далі вам буде представлено сторінка для побудови свого проекту з основними функціями та інструментами для роботи зі схемою та програмою. Веб-інтерфейс програмного забезпечення, представлений на рисунку 7.2, складається з таких основних блоків:

1. робоча область створення схеми
2. панель керування проектом
3. меню вибору відображення схеми (Breadbord View, Schematics View, PCB View)

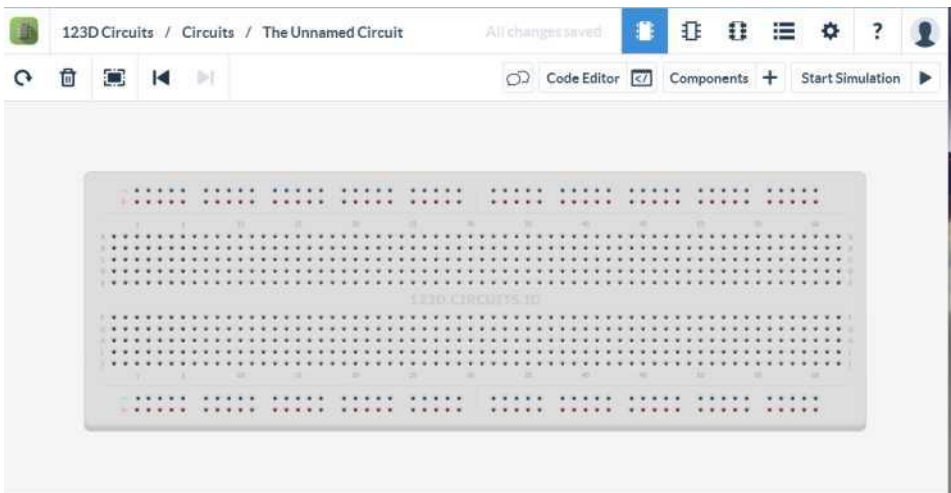


Рисунок 7.2 – Інтерфейс розробки проекту схеми

Для розуміння роботи в середовищі 123D Circuits, розробимо простий проект миготіння світлодіода. Створюємо схему за прикладом як зображено на рисунку 7.3.

Клікаємо на кнопку Components яка знаходиться на панелі керування проектом праворуч. У вікні компонентів яке відкрилося знаходимо плату Arduino Uno, обираємо її і розміщуємо її на робочій поверхні. Далі обираємо LED (світлодіод) і розміщуємо її на Breadbord. Відкривається вікно налаштування параметрів світлодіода, де можемо задати його ім'я та колір. Після встановлюємо резистор і також

задаємо його параметри: ім'я та опір 100 Ом. Як всі компоненти встановлені, з'єднуємо їх.

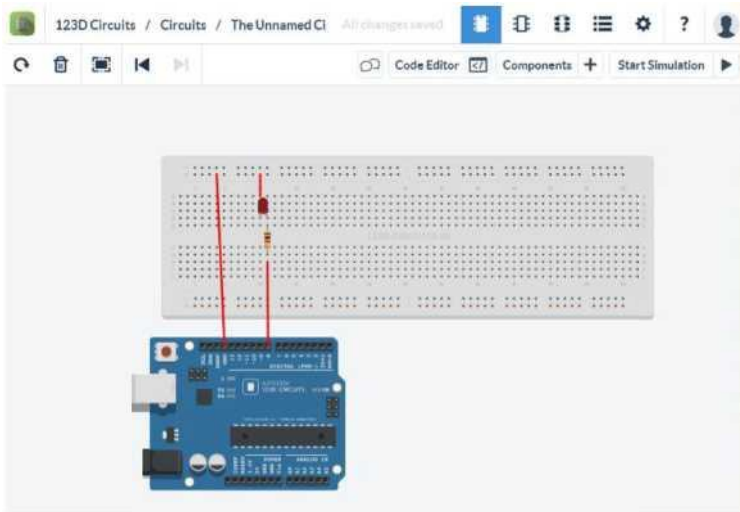


Рисунок 7.3 – Приклад схеми

Викличемо вікно редагування коду клікнувши на кнопку «Code Editor» яка знаходиться праворуч на панелі керування проектом. У редактор копіюємо код:

```
int led = 8;

void setup() {
  pinMode(led, OUTPUT);
}

void loop() {
  digitalWrite(led, HIGH); delay(1000);
  digitalWrite(led, LOW); delay(1000);
}
```

Компілюємо і завантажуюємо код клікнувши на кнопку «Upload&Run».

2. Завдання до лабораторної роботи

- 2.1 Розробіть у 123D Circuits проект мерехтіння 10 світлодіодів.
- 2.2 Розробіть у 123D Circuits будь-який проект з підключенням осцилографа.

2.3 Розробіть у 123D Circuits будь-який проект з підключеним мультиметром.

3. Зміст звіту

- 3.5. Тема та мета роботи
- 3.6. Три схеми проекту: компонентна, принципова, друкована.
- 3.7. Код програми
- 3.8. Висновки з роботи

ЛІТЕРАТУРА

1. Шегедин, О.І. Теоретичні основи електротехніки : навч.посіб. / О.І. Шегедин, В.С. Маляр.; Львів : Магнолія 2006, 2012. - 167с.
2. Коруд, В.І. Електротехніка : підручник для ВНЗ / В.І.Коруд, О.Є. Гамола.; за заг. ред. В.І. Коруда.- 3-тє вид., перероб. і доп.- Львів:Магнолія, 2007.- 447с.
3. Мілих, В.І. Електротехніка,електроніка та мікропроцесорна техніка:підручник для ВНЗ/В.І. Мілих, О.О.Шавьолкін;за ред.В.І.Мілих.-К.: Каравела, 2007.-688с.
4. Синдеев, Ю.Г. Электротехника с основами электроники: учебное пособие.; Ростов-на-Дону: Феникс, 2011. -407 с.
5. Arduino. [Електронний ресурс]. - Режим доступу: <https://www.arduino.cc/>
6. Autodesk 123D Circuit. [Електронний ресурс]. - Режим доступу: <https://123d.circuits.io/>
7. Atmel Studio. [Електронний ресурс]. Microchip Atmel. - Режим доступу: <http://www.atmel.com/tools/atmelstudio.aspx>